What is claimed is:

1. A processor which performs data processings including a plurality of execution units, comprising:

a storage which stores data used for processings of the execution units and processing results by the execution units, by each of the execution units;

a data processing part configured to acquire data of the execution units from said storage to perform the processings, and configured to output the processing results in said storage;

an execution unit judgement part configured to determine whether or not said storage holds data used for the processings of a certain execution unit, and whether or not said storage has a vacant region capable of storing the processing result of the certain execution unit; and

an execution unit determination part which determines an execution unit to be processed next among said plurality of execution units, based on a result judged by said execution unit judgement part.

2. The processor according to claim 1,

wherein said storage includes:

a first storing part which stores data used for the processing of the execution unit prescribed in advance; and

a second storing part which stores the processing result obtained by performing the processings of the corresponding execution unit by using data acquired from said first storage, and stores data used by an other execution unit using the processing result by the corresponding execution unit,

wherein said execution unit judgement part judges whether or not said first storing part holds data used for the processing of the execution unit, and whether or not said second storage part has the vacant region storing the processing result of the execution unit.

3. The processor according to claim 2, further comprising a priority setting part configured to set priorities to said plurality of execution units,

wherein said execution unit determination unit determines the execution unit to be started up, based on the priorities set by said priority setting unit.

4. The processor according to claim 3, further comprising a start-up frequency measuring part configured to measure start-up frequencies of said plurality of execution units,

wherein said priority setting part sets the priorities based on the start-up frequency measured by said start-up frequency measuring part.

5. The processor according to claim 3, wherein when a first execution unit is being started up, said execution unit determination part holds data used for a second execution unit with a priority higher than said first execution unit, and suspends operation of said first execution unit to start up said second execution unit when said second storage has the vacant region to store the processing result by said second execution unit.

6. The processor according to claim 2, wherein said data processing part includes:

a register set group which has a plurality of register sets including all of information inherent to the execution units;

a register set selector which selects the register set used by the execution units determined by said execution unit determination part; and

an operation processing part configured to perform processing of the execution unit decided by said execution unit determination part by using the register set selected by said register set selector.

7. The processor according to claim 2, further comprising:

a register set group which has a plurality of register sets including all of information inherent to the execution units;

a register selector which selects the register set used by the execution unit determined by said execution unit determination part;

an operation processing part configured to perform processing of the execution units determined by said execution unit determination part by using the register set selected by said register set selector;

an external register set storage capable of storing contents of an arbitrary register set included in said register set group; and

a storage controller which transfers contents of the register set used by the other execution unit under suspension or to be suspended, included in the register set group, to said external register set storage, when contents of

the register set used by the execution unit is stored in said external register set storage, and transfers data from said external register set storage to the register set.

8. The processor according to claim 7, further comprising a hit determination part configured to determine whether or not the execution unit determined by said execution unit determination part exists in the register set group,

wherein said register set selector selects the register set from said register set group when said hit determination part determines that the register set used by the execution unit exists in said register set group, and selects the register set which holds data transferred from said external register set storage when said hit determination part determines that the register set does not exist in said register set group.

9. The processor according to claim 8, further comprising a retreating register set determination part configured to select the register set to retreat into said external register set storage from said register set group when said hit determination part determines that the register set does not exist in said register set group,

wherein said storage controller transfers contents of the register set selected by said retreating register set determination part to said external register set storage.

10. The processor according to claim 6, wherein said execution unit determination part determines the execution unit by taking into consideration the judgement result by said execution unit judgement part, and even when said storage controller is performing an exchanging processing of the register sets between said register set group and said external register set storage, determines the execution unit capable of executing the processings without being influenced on the exchanging processing.

11. The processor according to claim 6, wherein said storage controller transfers only a value of the register changed among said register set group, to said external register set storage when there is a necessity to update contents stored in said external register set storage.

12. The processor according to claim 6, wherein said storage controller transfers only contents of the registers changed by executing an other execution unit from said external register set storage to said register set group, when it is necessary to transfer data from said external register set storage to said register set group.

13. The processor according to claim 6, wherein said storage controller transfers contents of the register set to said external register set storage as far as the same register set as the register set stored in said external register set storage does not exist in said register set group.

14. An arithmetic operation processing method, comprising:

executing processings by each of execution units which executes time-sharing processings for a certain data processing;

storing data used by processing for the execution unit prescribed in advance into a first storage;

storing the processing result obtained by processing of the corresponding execution unit by using data acquired from said first storage into a second storage, and storing data used by processings of an other execution unit using the stored processing result by the corresponding execution unit;

judging whether or not said first storage holds data using to the processings of the execution unit, and whether or not said second storage has a vacant region to store the processing result of the execution unit; and

determining the execution unit to be started up next among said plurality of execution units.

15. A processor, comprising:

a data processing part configured to execute processings by each of execution units which execute time-sharing processings for a certain data processing;

a plurality of storages which stores data used by the execution unit to be executed by said data processing part, or an execution result of data used by the execution unit to be executed by said data processing part; and

a priority determination part configured to determine priorities of the

execution units using data stored in the storages based on the amount of data stored in said plurality of storages.

16. The processor according to claim 15, wherein said priority determination part determines a higher priority for the execution unit receiving data to be processed from said storage, as the amount of data stored by a certain storage is larger.

17. The processor according to claim 15, wherein said priority determination part determines a lower priority for the execution unit storing the execution result in said storage, as the amount of data stored by a certain storage is larger.

18. The processor according to claim 16, wherein as compared a first case where the amount of data stored in the storage receiving data to be processed by a certain execution unit is in an increasing tendency with a second case where data stored in the storage receiving data to be processed by the certain execution unit is in a decreasing tendency, even if the amount of data stored in said storage in said first case is equal to the amount of data stored in said storage in said second case, the priority of the execution unit receiving data to be processed from said storage in said second case is set higher than that of the execution unit in said first case.

19. The processor according to claim 17, wherein as compared a first case where the amount of data stored in the storage storing the execution result by a certain execution unit is in an increasing tendency with a second case where the amount of data stored in the storage storing the execution result by the certain execution unit is an decreasing tendency, even if the amount of data stored in said storage in said first case is equal to the amount of data stored in said storage in said second case, the priority of the execution unit storing the execution result in the storage in said first case is set higher than that of the execution unit in the second case.

20. The processor according to claim 15, wherein said priority determination part sets the highest priority for the execution unit which receives data to be processed from a certain storage when determined that

the data amount stored by said certain storage exceeds a limit of memory capacity of said certain storage.

21. The processor according to claim 15, wherein said priority determination unit sets the lowest priority for the execution unit which receives data to be processed from a certain storage when determined that the data amount stored by said certain storage exceeds a limit of memory capacity of said certain storage.

22. A priority determination method, comprising:

executing processing by each of execution units which executes time-sharing processings for a certain data processing;

storing data used by an execution unit to be executed or an execution result by the execution unit, into a plurality of storages; and

determining a priority of the execution unit using data stored in said storages based on the data amount stored in said plurality storages.

23. The processor according to claim 16, wherein the execution units include a first execution unit and a second execution unit,

wherein the plurality of storages include a first storage to store data used by the first execution unit and a second storage to store data used by the second execution unit,

wherein the priority determination part sets the priority of the first execution unit lower than the second execution unit, if the following conditions are fulfilled:

(a) the amount of data of the first storage is equal to that of the second storage;

(b) the amount of data of the first storage is increasing tendency; and

(c) the amount of data of the second storage is decreasing tendency.

24. The processor according to claim 17, wherein the execution units include a first execution unit and a second execution unit,

wherein the plurality of storages include a first storage to store the execution result of the first execution unit and a second storage to store the execution result of the second execution unit,

wherein the priority determination part sets the priority of the first execution

unit higher than the second execution unit, if the following conditions are fulfilled:

(a) the amount of data of the first storage is equal to that of the second storage;

(b) the amount of data of the first storage is increasing tendency; and

(c) the amount of data of the second storage is decreasing tendency.